UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/531,397	03/21/2000	Joseph C. Ballantyne	3797.81466	6866

28319 7590 02/13/2004

BANNER & WITCOFF LTD.,
ATTORNEYS FOR MICROSOFT
1001 G STREET , N.W.
ELEVENTH STREET
WASHINGTON, DC 20001-4597

EXAMINER	
ALI, SYED J	
ART UNIT	PAPER NUMBER
2127	

DATE MAILED: 02/13/2004 13

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/531,397	Applicant(s) BALLANTYNE, JOSEPH C.	
	Examiner Syed J Ali	Art Unit 2127	

-- The MAILING DATE of this communication appears on the cover sh et with the correspond nce address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *29 December 2003*.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-9,11,13-19,21-25 and 27-30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-9,11,13-19,21-25 and 27-30* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on December 29, 2003 has been entered.

2. This office action is in response to Amendment B, paper number 12, which was filed December 29, 2003. Claims 1-9, 11, 13-19, 21-25, and 27-30 are presented for examination.

3. The text of those sections of Title 35, U.S. code not included in this office action can be found in a prior office action.

Claim Objections

4. Claim 9 is objected to under 37 CFR 1.75(c), as being of improper dependent form for failing to further limit the subject matter of a previous claim. Applicant is required to cancel the claims, or amend the claims to place the claims in proper dependent form, or rewrite the claims in independent form.

Claim 9 recites the limitation of "the counter issues a first interrupt to the interrupt controller in order to instruct the interrupt controller to allocate the CPU resources", which is broader than the limitation of "instructing an interrupt controller, via non-maskable interrupts from the counter, to allocate the CPU resources" recited in its parent claim 1.

[redacted]

Claim Rejections - 35 USC § 103

5. Claims 1-7, 9, 11, 16-19, 21, 23-25, 27-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gulick (previously cited) in view of Nakamura (USPN 5,437,047) in view of Reiffin (previously cited).

As per claim 1, Gulick discloses a method of scheduling CPU resources comprising the steps of:

using a counter to determine when to allocate the CPU resources (col. 5 lines 14-21, "Timer 228 provides a real-time hardware interrupt to CPU 102", wherein Applicant's specification indicates that a timer can be used as the claimed counter [pg. 4]); and

instructing the CPU to allocate resources in real-time (col. 4 lines 49-67, "The peripheral expansion bus is used for connecting various peripherals to the computer system, including an interrupt system, a real time clock [RTC] and timers", wherein these particular components drive the CPU allocation, and are implemented in real time).

Nakamura discloses the following limitations not shown by Gulick, specifically allocating CPU resources via non-maskable interrupts issued from the counter (col. 3 lines 7-37, "Each of the processors 10-1 and 10-2 is provided with a NMI timer interrupt unit 11 which generates a non-maskable timer interrupt...periodically at a predetermined time interval").

It would have been obvious to one of ordinary skill in the art to combine Gulick with Nakamura since the disclosure of Gulick fails to specify the type of interrupt that is issued by the timer. Although Gulick does disclose the use of non-maskable interrupts, these interrupts are issued from the operating system such that an executing task may be preempted by a higher priority task (Abstract, "an isochronous task may be preempted to execute a higher priority task"; col. 11 lines 3-23, "If the task does not complete within the specified maximum duration, termination unit 224 may invoke an interrupt to terminate the task. In one embodiment, the interrupt invoked is a non-maskable interrupt"). The function of the non-maskable interrupt issued by the operating system is similar to the interrupt issued by the timer, i.e., to terminate or preempt the executing task in favor of another task or a higher priority task. Thus, it would be beneficial to specify that the interrupt issued by the timer is also non-maskable, such that it is guaranteed that the interrupt service routine is executed. Thus, the combination of Gulick and Nakamura provides issuing a non-maskable interrupt from a timer (or counter) to preempt an executing task.

Reiffin discloses the following limitations not shown by the modified Gulick, specifically instructing an interrupt controller to allocate the CPU resources (col. 4 lines 1-13, "When the end of the current timeslice is thereby determined by the counter the [counter] then signals the interrupt controller which in turn activates the interrupt input of the CPU...to initiate an interrupt operation").

It would have been obvious to one of ordinary skill in the art to combine the modified Gulick with Reiffin since both Gulick and Nakamura disclose issuing an interrupt from a timer (or counter) directly to the interrupt input of the CPU. This limits the specific functionality that could potentially be implemented within the interrupt controller. Specifically, Gulick discloses an operating system issuing a non-maskable interrupt to the CPU in order to preempt a lower priority task with a higher priority task. The reason for this is that the determination as to which task is of higher priority must be made within some sort of software module. Since the interrupt issued within Gulick is strictly a hardware interrupt, such a determination cannot be made satisfactorily. Thus, to use an interrupt controller, as disclosed by Reiffin, would allow the interrupt service routine to be implemented within the interrupt controller. Such a modification would be a marked improvement over the modified Gulick, since all interrupts could be issued from the timer.

As per claim 2, Gulick discloses the method of claim 1 wherein only a portion of the CPU resources are allocated (col. 7 lines 3-29, "the portion of the operating system bandwidth allocated to isochronous tasks may be limited").

As per claim 3, Gulick discloses the method of claim 1 wherein all of the CPU resources are allocated (col. 7 lines 3-29, "a user via user interface 220 may increase the percentage of the operating system bandwidth allocated to isochronous tasks", wherein the percentage could conceivably be adjusted such that the entirety of the operating resources are allocated to isochronous tasks).

As per claim 4, Gulick discloses the method of claim 2 wherein the CPU resources are allocated to at least one thread, and the CPU resources are allocated by determining a duration of time (col. 6 lines 8-17, "Scheduler 218 may additionally output a signal to termination module 224 indicating the duration of an isochronous task") and a periodicity for execution of said at least one thread (col. 5 lines 14-21, “the period of the time-slice interrupt is variable based on the interval of the currently executing isochronous tasks”).

As per claim 5, Gulick discloses the method of claim 3 wherein the CPU resources are allocated to at least one thread, and the CPU resources are allocated by determining a duration of time (col. 6 lines 8-17, “Scheduler 218 may additionally output a signal to termination module 224 indicating the duration of an isochronous task”) and a periodicity for execution of said at least one thread (col. 5 lines 14-21, “the period of the time-slice interrupt is variable based on the interval of the currently executing isochronous tasks”).

As per claim 6, Gulick discloses the method of claim 1 wherein the counter is a performance counter (col. 5 lines 14-43, “system clock 226 is coupled to timer 228. System clock 226 provides a stable clock signal to timer 228”, wherein Applicant states on pg. 13 “that the performance counter 200 could be any type of programmable or re-settable counter”, including counting a clock cycle).

As per claim 7, Gulick discloses the method of claim 6 wherein the performance counter counts machine cycles in order to determine when to allocate the CPU resources (col. 5 lines 14-43, “timer 228 divides the system clock down to a lower frequency. For example, if the system clock is an 8 kHz clock, timer 228 may divide system clock 226 by eight to create a one millisecond interrupt”, wherein the timer can be adjusted to the system clock according to the user’s preference).

As per claim 9, Reiffin discloses the method of claim 1 wherein the counter issues a first interrupt to the interrupt controller in order to instruct the interrupt controller to allocate the CPU resources (col. 4 lines 1-13, "When the end of the current timeslice is thereby determined by the counter the [counter] then signals the interrupt controller which in turn activates the interrupt input of the CPU...to initiate an interrupt operation").

It would have been obvious to one of ordinary skill in the art to combine Gulick, Nakamura, and Reiffin for reasons discussed above in reference to claim 1.

As per claim 11, Nakamura discloses the method of claim 9 wherein the first interrupt is non-maskable (col. 3 lines 7-37, "Each of the processors 10-1 and 10-2 is provided with a NMI timer interrupt unit 11 which generates a non-maskable timer interrupt...periodically at a predetermined time interval").

It would have been obvious to one of ordinary skill in the art to combine Gulick, Nakamura, and Reiffin for reasons discussed above in reference to claim 1.

As per claim 16, Gulick discloses a method of scheduling resources on at least one microprocessor that includes at least one performance counter, and at least one CPU, said method comprising the steps of:

allowing the CPU to execute a first thread (col. 9 line 44 - col. 10 line 25, "Scheduler 218 determines that Task B is a quick-slice task scheduled to be executed. Accordingly, scheduler

218 preempts Task A which is a standard slice and passes control to Task B", wherein Task A is the "first" thread);

using the performance counter to determine when to allocate the resources to a second thread on a real-time basis (col. 5 lines 14-21, "Timer 228 provides a real-time hardware interrupt to CPU 102", wherein Applicant's specification indicates that a timer can be used as the claimed counter [pg. 4]; col. 4 lines 49-67, "The peripheral expansion bus is used for connecting various peripherals to the computer system, including an interrupt system, a real time clock [RTC] and timers", wherein these particular components drive the CPU allocation, and are implemented in real time);

issuing a first interrupt from the from the performance counter to the CPU when it is time to allocate the resource to the second thread (col. 5 lines 14-21, "Timer 228 provides a real-time hardware interrupt to CPU 102");

instructing the CPU to switch execution from the first thread to the second thread (col. 9 line 44 - col. 10 line 25, "Scheduler 218 determines that Task B is a quick-slice task scheduled to be executed. Accordingly, scheduler 218 preempts Task A which is a standard slice and passes control to Task B");

instructing the CPU to stop execution of the first thread (col. 9 line 44 - col. 10 line 25, "Scheduler 218 determines that Task B is a quick-slice task scheduled to be executed. Accordingly, scheduler 218 preempts Task A which is a standard slice and passes control to Task B"); and allocating resources to the second thread (col. 9 line 44 - col. 10 line 25, "Scheduler 218 determines that Task B is a quick-slice task scheduled to be executed. Accordingly, scheduler 218 preempts Task A which is a standard slice and passes control to Task B").

Nakamura discloses the following limitations not shown by Gulick, specifically that the interrupts are non-maskable (col. 3 lines 7-37, "Each of the processors 10-1 and 10-2 is provided with a NMI timer interrupt unit 11 which generates a non-maskable timer interrupt...periodically at a predetermined time interval");

the CPU storing first current state information regarding execution of the first thread (col. 3 lines 7-37, "Each of processors also include a context information collecting unit 12 for fetching and storing information concerning the program running state of a program loaded in the associated processor. Further, one of the processors 10-1 and 10-2...is provided with an accumulated data output unit 13 for outputting or transferring the program run information to a file 16 which serves as an external storage medium"); and

the CPU restoring second current state information regarding execution of the second thread (col. 3 lines 38-46, "Each of the context information collecting facilities 12 may include by a handler program...activated by the interrupt issued by NMI timer interrupt unit 11 and serving to fetch context information").

Reiffin discloses the following limitations not shown by either Gulick or Nakamura, specifically that the microprocessor includes at least one programmable interrupt controller, wherein the performance counter issues interrupts to the programmable interrupt controller when it is time to allocate CPU resources to the second thread (col. 4 lines 1-13, "When the end of the current timeslice is thereby determined by the counter the [counter] then signals the interrupt controller which in turn activates the interrupt input of the CPU...to initiate an interrupt operation"); and

instructing the programmable interrupt controller to issue the second interrupt to the CPU that instructs the CPU to switch execution from the first thread to the second thread (col. 4 lines 1-13, "When the end of the current timeslice is thereby determined by the counter the [counter] then signals the interrupt controller which in turn activates the interrupt input of the CPU...to initiate an interrupt operation").

It would have been obvious to one of ordinary skill in the art to combine Gulick, Nakamura, and Reiffin for reasons discussed above in reference to claim 1.

As per claim 17, Reiffin discloses the method of claim 16 wherein the programmable interrupt controller is an APIC. The discussion of claim 16 discusses the programmable interrupt controller, and an APIC is defined to be a programmable interrupt controller that can handle interrupts from and for multiple CPUs. Therefore, since Reiffin discloses that is for use in a multiprocessor system (Abstract, "The computer time and processing power which would otherwise be wasted while waiting for slow input/output operation is instead utilized to provide a powerful parallel multiprocessor system for handling compute-intensive tasks too large for an individual workstations"), Reiffin inherently discloses that the programmable interrupt controller is an APIC.

It would have been obvious to one of ordinary skill in the art to combine Gulick, Nakamura, and Reiffin for reasons discussed above in reference to claim 1.

As per claim 18, Gulick discloses the method of claim 17 wherein the microprocessor is selected from the group consisting of: a Pentium 4GB, a Pentium Pro 64GB, a Pentium MMX 4GB MMX, a Pentium II 4GB MMX, a Pentium II 4GB MMX KNI, a Celeron 4GB MMX, a Xeon PII 64GB MMX and a Xeon PIII 64GB MMX KNI (col. 4 lines 19-32, "The chipset 106 includes arbitration logic 107 as shown. In one embodiment, the chipset is the Triton chipset available from Intel Corporation", wherein the disclosure is easily modifiable to include any of a number of chipsets, and as the claimed microprocessors are all of the Intel family of processors, the disclosure of Gulick therein satisfies this limitation).

As per claim 19, Gulick discloses a computer-readable medium having computer-executable instructions stored for performing steps comprising:

using a scheduler to control execution of at least one thread based on an interrupt (col. 9 line 44 - col. 10 line 25, "Scheduler 218 determines that Task B is a quick-slice task scheduled to be executed. Accordingly, scheduler 218 preempts Task A which is a standard slice and passes control to Task B", wherein Task A is the "first" thread); and

using at least one counter to issue a first interrupt to the CPU to notify the scheduler when to switch execution of said at least one thread on a real-time basis (col. 5 lines 14-21, "Timer 228 provides a real-time hardware interrupt to CPU 102", wherein Applicant's specification indicates that a timer can be used as the claimed counter [pg. 4]; col. 4 lines 49-67, "The peripheral expansion bus is used for connecting various peripherals to the computer system, including an interrupt system, a real time clock [RTC] and timers", wherein these particular components drive the CPU allocation, and are implemented in real time).

Nakamura discloses the following limitations not shown by Gulick, specifically that the interrupts are non-maskable (col. 3 lines 7-37, "Each of the processors 10-1 and 10-2 is provided with a NMI timer interrupt unit 11 which generates a non-maskable timer interrupt...periodically at a predetermined time interval").

Reiffin discloses the following limitations not shown by either Gulick or Reiffin, specifically that the interrupts are issued by an interrupt controller, wherein the interrupt controller relays the interrupt to the scheduler to switch execution of threads (col. 4 lines 1-13, "When the end of the current timeslice is thereby determined by the counter the [counter] then signals the interrupt controller which in turn activates the interrupt input of the CPU...to initiate an interrupt operation").

It would have been obvious to one of ordinary skill in the art to combine Gulick, Nakamura, and Reiffin for reasons discussed above in reference to claim 1.

As per claim 21, Gulick discloses the computer-readable medium of claim 19 wherein said least one counter is a performance counter and counts CPU cycles (col. 5 lines 14-43, "system clock 226 is coupled to timer 228. System clock 226 provides a stable clock signal to timer 228", wherein Applicant states on pg. 13 "that the performance counter 200 could be any type of programmable or re-settable counter", including counting a clock cycle, "timer 228 divides the system clock down to a lower frequency. For example, if the system clock is an 8 kHz clock, timer 228 may divide system clock 226 by eight to create a one millisecond interrupt", wherein the timer can be adjusted to the system clock according to the user's preference).

As per claim 23, the modified Gulick does not specifically disclose the computer-readable medium of claim 20 further comprising instructions for executing said at least one thread at a highest IRQ level. However, "Official Notice" is taken that it would have been obvious to one of ordinary skill in the art to place the interrupting thread at a highest IRQ level since that would ensure that the interrupting task is serviced without being preempted by another task. To that end, threads of a highest priority may be guaranteed Quality of Service, such that tasks of a highest priority are serviced accordingly.

As per claim 24, Reiffin discloses the computer-readable medium of claim 20 further comprising instructions for executing said at least one thread in a transparent manner so that at least one operating-system process is unaware of the execution of said at least one thread (col. 4 lines 41-59, "control of the CPU is rapidly switched back and forth between a local task and a network task so rapidly that the two tasks are said to be running concurrently", wherein the local task could be considered the operating system thread and can be treated simply as a permanent process, and does not need to be made aware of other background processes).

It would have been obvious to one of ordinary skill in the art to combine Gulick, Nakamura, and Reiffin for reasons discussed above in reference to claim 1.

As per claim 25, Gulick discloses the computer-readable medium of claim 24 further comprising instructions for executing all of said operating-system process and all of said at least one threads as a single real-time thread (col. 1 line 48 - col. 2 line 35, "Several partially effective methods of providing real-time capability to general-purpose operating systems have been used", wherein Gulick discloses how a real-time interrupt can be used within a general purpose operating system to execute all processes therein as a single real-time process).

As per claim 27, Gulick discloses the computer-readable medium of claim 19 further comprising instructions for allocating at least a portion of a CPU's resources to an operating-system process and using the remaining CPU resources for execution of said at least one thread (col. 7 lines 4029, "scheduler 218 may determine whether sufficient resources are available to service the tasks based upon the available portion of the operating system cycle for isochronous tasks and the resources used by other isochronous tasks. If sufficient resources are available, scheduler 218 schedules the isochronous tasks of the application", wherein the CPU resources are divided among the operating system bandwidth and the isochronous tasks to be executed).

As per claim 28, Gulick discloses the computer-readable medium of claim 27 further comprising instructions for releasing the CPU resources back to the operating-system process when said at least one thread finishes execution (col. 7 lines 18-29, "a user via user interface 220 may increase the percentage of the operating system bandwidth allocated to isochronous tasks", wherein once the task has been completed, the user may then reallocate the operating system bandwidth to give CPU resources back to the operating system).

As per claim 29, Gulick discloses the computer-readable medium of claim 27 further comprising instructions for releasing the CPU resources to another thread when said at least one thread finishes execution (col. 7 lines 18-29, "a user via user interface 220 may increase the percentage of the operating system bandwidth allocated to isochronous tasks", wherein once the task has been completed, the user may then reallocate the operating system bandwidth to give CPU resources back to the operating system).

As per claim 30, Gulick discloses the computer-readable medium of claim 19 further comprising instructions for allocating a predetermined number of CPU cycles for execution of an operating-system process and using the remaining CPU cycles for execution of said at least one thread (col. 7 lines 4029, "scheduler 218 may determine whether sufficient resources are available to service the tasks based upon the available portion of the operating system cycle for isochronous tasks and the resources used by other isochronous tasks. If sufficient resources are available, scheduler 218 schedules the isochronous tasks of the application", wherein the CPU resources are divided among the operating system bandwidth and the isochronous tasks to be executed).

6. Claims 8 and 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gulick in view of Nakamura in view of Reiffin in view of Patterson et al. (previously cited) (hereinafter Patterson).

As per claim 8, Patterson discloses the following limitations not shown by the modified Gulick, specifically the method of claim 6 wherein the performance counter counts executed computer instructions (col. 2 line 66 - 3 line 43, "an integer field, functioning as a global counter, is incremented based on a regularly occurring event", "Global counter 222 is incremented based on the output of clock 204, and may be, for example, incremented one 'tick' for every million cycles of clock 204" wherein this is merely one way in which the counter may be set up, and to align the clock such that a tick corresponds to the number of clock cycles necessary for a single instruction to issue is merely another embodiment of the same concept).

It would have been obvious to one of ordinary skill in the art to combine the modified Gulick with Patterson since within the framework of a real time application environment, restricting the interrupt controller to be triggered by only certain events prevents the environment from being modified to suit particular needs. For example, a real time task may be required to perform a certain amount of work before it is preempted, in addition to having a deadline. To that end, by preempting the task at the expiration of the period, rather than allowing the task to complete a minimum amount of work may cause performance of later tasks to suffer. Allowing the counter to be incremented based on any type of regularly occurring event, such as a computer instruction, allows the capabilities of the system to be modified to suit any number of specific needs.

As per claim 22, Patterson discloses the following limitations not shown by the modified Gulick, specifically the computer-readable medium of claim 19 wherein said at least one counter is part of a CPU and counts executed computer instructions (col. 2 line 66 - 3 line 43, "an integer field, functioning as a global counter, is incremented based on a regularly occurring event", "Global counter 222 is incremented based on the output of clock 204, and may be, for example, incremented one 'tick' for every million cycles of clock 204" wherein this is merely one way in which the counter may be set up, and to align the clock such that a tick corresponds to the number of clock cycles necessary for a single instruction to issue is merely another embodiment of the same concept).

It would have been obvious to one of ordinary skill in the art to combine the modified Gulick with Patterson for reasons discussed above in reference to claim 8.

7. Claims 13-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gulick in view of Nakamura.

As per claim 13, Gulick discloses a method of scheduling resources on at least one microprocessor that includes a CPU and a device, the method comprising the steps of:

using the device to determine, in response to a first interrupt, when to allocate the resources in real-time (col. 5 lines 37-43, "timer 228 divides the system clock down to a lower frequency. For example, if the system clock is an 8 kHz clock, timer 228 may divide system clock 226 by eight to create a one millisecond interrupt", wherein the first interrupt is essentially generated by the system clock, and is based on a real time clock);

causing the device to issue a second interrupt to the CPU when it is time to allocate the resources (col. 5 lines 14-21, "Timer 228 provides a real-time hardware interrupt to CPU 102"); and

causing the CPU to allocate the resources in response to the second interrupt (col. 5 lines 14-21, "Timer 228 provides a real-time hardware interrupt to CPU 102", wherein the interrupt issued to the CPU preempts the executing task and schedules another task).

Nakamura discloses the following limitations not shown by Gulick, specifically that the interrupts are non-maskable (col. 3 lines 7-37, "Each of the processors 10-1 and 10-2 is provided with a NMI timer interrupt unit 11 which generates a non-maskable timer interrupt...periodically at a predetermined time interval").

It would have been obvious to one of ordinary skill in the art to combine Gulick with Nakamura since the disclosure of Gulick fails to specify the type of interrupt that is issued by the timer. Although Gulick does disclose the use of non-maskable interrupts, these interrupts are issued from the operating system such that an executing task may be preempted by a higher priority task (Abstract, "an isochronous task may be preempted to execute a higher priority task"; col. 11 lines 3-23, "If the task does not complete within the specified maximum duration, termination unit 224 may invoke an interrupt to terminate the task. In one embodiment, the interrupt invoked is a non-maskable interrupt"). The function of the non-maskable interrupt issued by the operating system is similar to the interrupt issued by the timer, i.e., to terminate or preempt the executing task in favor of another task or a higher priority task. Thus, it would be beneficial to specify that the interrupt issued by the timer is also non-maskable, such that it is guaranteed that the interrupt service routine is executed. Thus, the combination of Gulick and Nakamura provides issuing a non-maskable interrupt from a timer (or counter) to preempt an executing task.

As per claim 14, Gulick discloses the method of claim 13 wherein the device is a performance counter (col. 5 lines 14-43, "system clock 226 is coupled to timer 228. System clock 226 provides a stable clock signal to timer 228", wherein Applicant states on pg. 13 "that the performance counter 200 could be any type of programmable or re-settable counter", including counting a clock cycle).

As per claim 15, Gulick discloses the method of claim 13 wherein the device is a timer (col. 5 lines 14-21, "Timer 228 provides a real-time hardware interrupt to CPU 102").

Response to Arguments

8. Applicant's arguments with respect to claims 1-9, 11, 13-19, 21-25, and 27-30 have been considered but are moot in view of the new grounds of rejection.

Conclusion

9. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Syed J Ali whose telephone number is (703) 305-8106. The examiner can normally be reached on Mon-Fri 8-5:30, 2nd Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai T An can be reached on (703) 305-9678. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Syed Ali
February 4, 2004

MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100